

A genetic algorithm rooted in integer encoding and fuzzy controller

M. Jalali Varnamkhasti

Department science, Isfahan Branch, Islamic Azad University, Isfahan, Iran

Article Info

Article history:

Received Feb 12, 2019

Revised Apr 12, 2019

Accepted May 1, 2019

Keywords:

Benchmark problems

Fuzzy controller

Genetic algorithm

Sexual selection

ABSTRACT

The premature convergence is the essential problem in genetic algorithms and it is strongly related to the loss of genetic diversity of the population. In this study, a new sexual selection mechanism which utilizing mate chromosome during selection proposed and then technique focuses on selecting and controlling the genetic operators by applying the fuzzy logic controller. Computational experiments are conducted on the proposed techniques and the results are compared with some other operators, heuristic and local search algorithms commonly used for solving benchmark problems published in the literature.

Copyright © 2019 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

M. Jalali Varnamkhasti,
Department science, Isfahan Branch,
Islamic Azad University, Isfahan, Iran.
Email; m.jalali@khuaisf.ac.ir

1. INTRODUCTION

Premature convergence is a significant problem for finding the optimal solution in Genetic Algorithm (GAs) and it is strongly related to the loss of genetic diversity of the population. If the diversity of the population is low, the GA will converge very quickly [1]. On the other hand, if the diversity of the population is too high, the GA will takes a lot of time to converge and this may caused wastage in computational resources [2]. A number of factors influence the genetic diversity of the population namely, population size, selection mechanism, crossover and mutation operators as well as their probabilities and replacement strategy [3].

Fuzzy Logic (FL) and GAs can be merged in two possible cases: (i) the application of GAs for solving optimization problems related with fuzzy systems and (ii) using fuzzy tools and FL-based techniques for modeling and adapting different GAs components [4]. The performance of the GA is strongly related to the equilibrium between exploration and exploitation. Exploration in GAs means inquiring into new and unknown region in a search space while exploitation refers to utilization of knowledge obtained from exploration in achieving better situations in the search space [4, 5]. Establishing a suitable relationship between exploitation and exploration during the GA implementation is critically important for preventing the premature convergence problem. Some researchers utilized FL by with passion control the parameters in GA in order to create suitable balance between exploitation and exploration [6, 7].

The objectives of this study are to establish and assess a new mechanism as well as the use of Fuzzy Logic Controller (FLC) for selecting and controlling the parameters in GAs. The specific objectives in this research to be addressed are:

- to propose a new technique for choosing mate chromosomes during sexual selection in a GA.
- to assess the diversity of the population based on its phenotype and genotype characteristics.

- to realize a membership function and setting linguistic labels to be used in FLC in order to control the parameters in GA.
- to explore the selection benchmark problems based on integer encoding.
- to perform extensive computational experiments on benchmark problem instances to determine the effectiveness of the proposed algorithms in generating near-optimal solutions.

2. LITERATURE REVIEW OF FUZZY GENETIC ALGORITHM

A management technique of crossover and mutation probabilities on the basis of Fuzzy Logic (FL) was proposed by Song et al. [8]. Li et al. [9] used fuzzy tools in GA for controlling the crossover and mutation probabilities. Their FGA utilized the information of both the whole generation and the particular chromosomes. Wang [10] designed a FGA based on population diversity measurements whereby the fuzzy controller was employed in controlling the crossover and mutation probabilities. Wang and Hong [11] utilized fuzzy systems for dynamically controlling the crossover and mutation probabilities during the optimization procedure.

An exhaustive analysis of fuzzy-adaptive GA was presented by Herrera and Lozano [12]. They depicted the stages for their design and incorporated taxonomy into their fuzzy-adaptive GA. The taxonomy technique constitutes the combination of two appearances: the level at which the adaptation takes place and the way how the rule bases are obtained. They considered different groups of taxonomies and consequently produced some fuzzy-adaptive GAs.

In another study conducted by Herrera and Lozano [5], an adaptive approach for controlling the mutation probability based on a utilization of FLCs was suggested. In this new approach, the measure for success of controlling the mutation probability was the advance produced by the GA over the last generation. Herrera et al. [13] presented two fuzzy tools to improve the GA behaviour: fuzzy connectives and fuzzy logic-based systems. In this technique fuzzy connective was utilized to design crossover operator for real coded GA. This technique has some shortcomings, among the rest:

- In each generation minimum and maximum value must be calculated and these numbers are compared to each pair chromosome, which are selected for crossover and is very time consuming.
- This method can be used only for crossover operators that replace same positions of genes.
- This method was tested only for a nonlinear number function and it was not challenged seriously.

Lee and Takagi [14] proposed a method for controlling the crossover rate, and mutation rate. They also presented an automatic fuzzy design technique which was based on GAs. This technique had two major defects:

- Only phenotype characteristic used as the input variable.
- Using the meta-level GA to find the optimized fuzzy system for the dynamic parameterized GA is computationally expensive.

Jalali and Lee [15] proposed a new technique for choosing the female chromosome during sexual selection in a GA. A bi-linear allocation lifetime approach is used to label the chromosomes based on their fitness value which will then be used to characterize the diversity of the population. The application of this technique is used by Jafari et al. [16] for Committee neural networks. Also in another work Jalali and Lee [17] introduced a genetic algorithm with fuzzy crossover operator and probability.

Yand and Jin [18] are presented a FGA such that chromosomes are included of quantum bits encoded by real digit. Odeh et al [16] presented a hybrid algorithm that combines Fuzzy Logic Controller (FLC) and Genetic Algorithms (GAs) and its application on a traf signal system. Pelusi et al. proposed some Gravitational Search Algorithm (GSA) based on fuzzy logic and some algorithms as GA, Particle Swarm Optimization (PSO) and Differential Evolution (DE), to improve GSA. A new technique of the FGA was designed to control the frequency of the system by Cam et al. [19]. This method showed a better act than the conventional and other modern control techniques.

2.1. Details of proposed algorithm

The algorithm of FGA in this study can be shown as follow:

begin

Initialise Population

Fitness Evaluation

repeat

Population Diversity

Sexual Selection

Fuzzy Controller

Fuzzy Crossover
 Fuzzy Mutation
 Replacement
until the end condition is satisfied
return the fittest solution found
end

2.2. Population diversity

Jalali and Lee [15] considered the genotype and phenotype properties for the measurement of the population diversity in a GA with fuzzy crossover operator and probability. In our study we are using the same strategy for the measurement of the population diversity. They introduced three parameters (1):

$$\begin{aligned}
 T_{1,t} &= \frac{n_{f_i}}{N} \\
 T_{2,t} &= \frac{f_{\max,t} - f_{avr,t}}{f_{\max,t}} \\
 T_{3,t} &= \frac{H(C_{f_{\max,t}} - C_{f_{\min,t}})}{L}
 \end{aligned} \tag{1}$$

where,

N =number of population.

n_{f_i} =number of different fitness values in generation t .

$f_{\max,t}$ =maximum fitness value in generation t .

$f_{avr,t}$ =average fitness value in generation t .

and $H(C_{f_{\max,t}} - C_{f_{\min,t}})$ is the Hamming Distance (HD) between the worst chromosome and the best one and L is the length of chromosome.

The $T_{1,t}$, $T_{2,t}$ and $T_{3,t}$ belong to the interval $[0, 1]$. If these numbers are close to zero, then the diversity of population is low and the algorithm has the potential to converge very soon. However, if it is near 1, the population shows a high level of diversity. As regards to $T_{1,t}$, $T_{2,t}$ and $T_{3,t}$ three membership functions are defined. The set of linguistic labels associated with $T_{1,t}$, and $T_{3,t}$, are low, medium and high. are low and high. The meanings of these labels are illustrated by Figure 1.

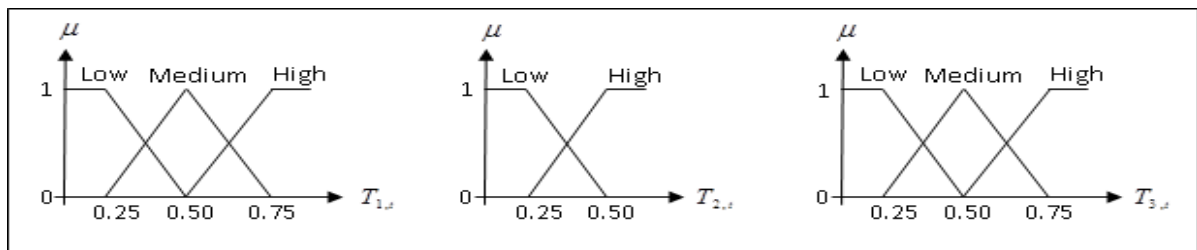


Figure. 1. The set of linguistic labels associated with $T_{1,t}$, $T_{2,t}$, and $T_{3,t}$ [15]

2.3. Sexual selection

When integer encoding is used, a wide variety of genes appear within the chromosomes. In this case, a relation between the fitness value and Hamming distance is proposed for sexual selection. Inspired by the non-genetic sex determination system prevalent in some reptile species where sex is determined by

the temperature at which the egg is incubated, the population are divided such that the male and female would be selected in an alternate way. The layout of the male and female chromosomes in each generation is different [15]. A female chromosome (female_chro) is selected by tournament selection size t from the female group. Then, the male_chro chromosome is selected, in order of preference based on:

- the maximum Hamming distance between the male chromosome and the female_chro, or
- the highest fitness value of male chromosome (if more than one male chromosome is having the maximum HD existed), or
- random selection.

2.4. Crossover

2.4.1. Fuzzy crossover operator selection

For selecting a crossover method based on integer encoding, some crossover methods introduced in literature are considered. Common abbreviations for these methods are presented in Table 1. Abbreviations of crossover operators.

Table 1. Abbreviation of crossover operators

Crossover methods	Abbreviation
Partially Mapped Crossover	PMX[16]
Order Crossover	OX[17]
Cycle Crossover	CX[18]
Position Based Crossover	PBX[19]
Alternating Position Crossover	APX[20]

When the genetic diversity of the population is high ($T_{3,t} \geq 0.50$), the PMX and OX crossover methods perform well. The CX, PBX and APX are suitable methods as well, but time consumption in the PMX and OX methods is less than in the CX, and when the genetic diversity of the population is high it is better to keep the genes' positions. On the other hand, if we use the PBX or APX method when the genetic diversity of the population is high, the GA may need a lot of time to converge, hence resulting in wastage of computational resources.

The genetic diversity of the chromosomes is medium ($0.25 \leq T_{3,t} \leq 0.50$) when some gene locations in the chromosomes are the same. In such case, the PMX and CX methods are not an excellent choice because the offspring and parent are identical or very similar to other whereas the OX, PBX or APX method is more appropriate for these types of chromosomes as shown in Figure 2.

On the other hand, when the genetic diversity of the population is low ($T_{3,t} \leq 0.25$), some chromosomes are similar and then the PMX, CX and PBX methods are not useful because the offspring and parent are identical or resemble one another. Under such condition the APX and OX methods are more suitable as shown in Figure 3.

Therefore, we can say that OX and APX operators are appreciate in all cases of population diversity, PBX is not useful when chromosomes are identical and PMX is not useful when genetic diversity of chromosomes is medium or low. CX is useful for high diversity but time consumption is high and if this operator is compared to other operators in a short computation time the results of this operator will be weaker than the other operators. Regarding to the above explanation of relative crossover operator and population diversity in integer encoding, three levels for crossover operators based on integer encoding shown in Table 2 are introduced.

We referred them as Low, Medium and High. Low Crossover Ability (CA) means effect of this operator on genetic diversity is low, and this is relative to the structure of crossover operator or time consumption. Medium CA means this operator can increase genetic diversity but not a lot. When is said CA is high means effect of this operator on genetic diversity is high.

Table 2. Categories of crossover operator based on crossover ability

Low	Medium	High
CX		OX
PMX	PBX	APX

Parent	CrossoverMethod	Offspring
Parent1: 1 3 4 9 5 6 7 8 2 10	PMX	Offspring1: 1 3 4 9 5 6 7 8 2 10
		Offspring2: 4 2 1 8 5 6 7 9 3 10
Parent2: 4 2 1 8 5 6 7 9 3 10	OX	Offspring1: 4 2 1 8 5 6 7 9 3 10
		Offspring2: 1 3 4 9 5 6 7 8 2 10
HD = 6	CX	Offspring1: 1 3 4 9 5 6 7 8 2 10
		Offspring2: 4 2 1 8 5 6 7 9 3 10
	PBX	Offspring1: 1 2 4 8 5 6 7 8 2 10
		Offspring2: 4 3 1 9 5 6 7 9 3 10
	APX	Offspring1: 1 4 3 2 9 8 5 6 7 10
		Offspring2: 4 1 2 3 8 9 5 6 7 10

Figure 2. Comparison of crossover methods for medium genetic diversity

Parent	CrossoverMethod	Offspring
Parent1: 1 3 4 9 5 6 7 8 2 10	PMX	Offspring1: 1 3 4 9 5 6 7 8 2 10
		Offspring2: 1 3 2 9 5 6 7 8 4 10
Parent2: 1 3 2 9 5 6 7 8 4 10	OX	Offspring1: 1 3 2 9 5 6 7 8 4 10
		Offspring2: 1 3 4 9 5 6 7 8 2 10
HD = 2	CX	Offspring1: 1 3 4 9 5 6 7 8 2 10
		Offspring2: 1 3 2 9 5 6 7 8 4 10
	PBX	Offspring1: 1 3 4 9 5 6 7 8 2 10
		Offspring2: 1 3 2 9 5 6 7 8 4 10
	APX	Offspring1: 1 3 2 9 5 6 7 8 4 10
		Offspring2: 1 3 4 9 5 6 7 8 2 10

Figure 3. Comparison of crossover methods for low genetic diversity

2.4.2. Fuzzy crossover probability selection

Crossover is based on the probabilistic model-building procedure of the extended compact GA. The probability of crossover is an effective indicator of premature convergence regain. The selection of crossover probability (P_c) critically affects the behaviour and performance of the GA. However, guiding selection principles exist in the literature, (e.g., Grefenstette [14], De Jong [12, 20], Goldberg and Sastry [21] and Schaffer [22]).

Researchers usually use a probability of 50%–100% for executing crossover [20]. Goldberg and Sastry [23] generalized the schema theorem for P_c . They argued that the selection pressure corresponds to disruption of schema and showed that when the building blocks are compact (its genes are located close to each other in the chromosome string), the GA works well for a wide range of combinations of P_c and selection rates.

Adaptive techniques that alter the probability of applying an operator in proportion to the observed performance of the chromosomes created by that operator in the course of a run were proposed by Davis [24]. Fernandes et al. [25] considered a GA that adapts the reproduction rate to the size of the population size under investigation.

In this study, the crossover probability is varied on the basis of the phenotype and genotype characteristics of the chromosome population. In addition, the population diversity is considered and its probability is estimated by a fuzzy logic controller. As to the $T_{1,t}=T_{2,t}$ and $T_{3,t}$ inputs, the set of linguistic labels associated with crossover probability comprises the descriptions low, medium and high. For each linguistic term, there is a triangular fuzzy set that defines its semantic (meaning) as shown in Figure 4.

2.5. Mutation

2.5.1. Fuzzy mutation operator selection

The arguments presented in this section led to the idea of using fuzzy system for selecting an appropriate mutation operator in order to obtain a more efficient mutation strategy. The proposed techniques will be categorized on the basis of the type of encoding each employs. In the case of integer encoding, many different mutation operators have been designed. However, unlike the mutation in binary encoding which only introduce small changes to the offspring, the mutation operators in integer encoding often modify the offspring greatly. Though, there are some issues concomitant to these methods when applied to problem like the p-median problem, p-hub, and incapacitated warehouse location problem. The reason behind these issues is the way how chromosomes are formed.

In these problems, chromosomes are created by considering the index for facilities. For instance, in the p-median problem where there are 50 facilities and $p_c=10$, the length of chromosomes is considered to

be 10. These 10 numbers are selected from index 1 to 50 and the layout of them is not important. Since the mutation operators are based on relocation of gene positions, selection mutation operator by fuzzy logic are not usable for the tackling the foregoing problems and accordingly in this thesis only pairwise mutation and Hypermutation [26] are employed. With this in mind, the FLC is not applied in the mutation operator selection since is only one mutation operator will be used.

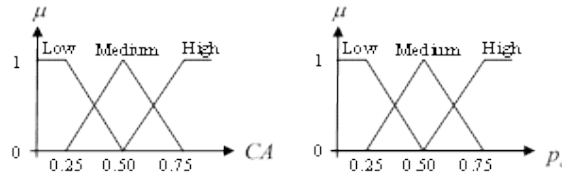


Figure 4. The set of linguistic labels [13]

2.5.2. Fuzzy mutation probability selection

As to the $T_{1,t}$, $T_{2,t}$ and $T_{3,t}$ inputs, the set of linguistic labels associated with mutation probability comprises the descriptions low, medium and high. For each linguistic term, there is a triangular fuzzy set that defines its semantic, i.e., meaning as shown in Figure 5.

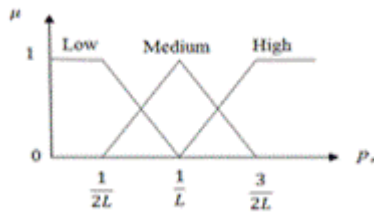


Figure 5. The set of linguistic labels associated P_m

In the traditional GA usually a very small number is used for mutation probability. Most of them used $P_m = \frac{1}{L}$, where L is the length of the chromosome. In the proposed technique, $P_m = \frac{1}{L}$ is used as the center point for the mutation probability. The range that is considered in this study is at the interval of $\left(\frac{1}{L} - \frac{1}{2L}, \frac{1}{L} + \frac{1}{2L}\right) = \left(\frac{1}{2L}, \frac{3}{2L}\right)$.

2.6. Fuzzy rules

The linguistic rules describing the control system consist of two parts: an antecedent block (between the **IF** and **THEN**) and a consequent block (following **THEN**). By making this type of evaluation, fewer rules can be evaluated, thus simplifying the processing logic and perhaps even improving the fuzzy logic system performance. Each rule has the possibility of generating a single rule for each output variable. In this paper, the inputs are combined logically using the AND/OR operator to produce output ($x_i, \mu_i(CA)$) and ($y_i, \mu_i(P_c)$) and ($z_i, \mu_i(P_m)$) response values for all the expected inputs (2-4) [24]:

$$\mu_i(CA) = \max \{ \mu_i(T_{1,t}), \{ \mu_i(T_{2,t}), \{ \mu_i(T_{3,t}) \} \} \} \quad (2)$$

$$\mu_i(P_c) = \min \{ \mu_i(T_{1,t}), \{ \mu_i(T_{2,t}), \{ \mu_i(T_{3,t}) \} \} \} \quad (3)$$

$$\mu_i(P_m) = \min \{ \mu_i(T_{1,t}), \{ \mu_i(T_{2,t}), \{ \mu_i(T_{3,t}) \} \} \} \quad (4)$$

where $i=1,2,\dots, \text{number of rules}$

Therefore, the proposed fuzzy system with three input variables will have 18 rules for each output variable (CA , P_c and P_m). This fuzzy rule base is collectively presented in Table 3. The fuzzy outputs for all

rules are finally aggregated to one fuzzy set. To obtain a crisp decision from this fuzzy set, we use the Centre of Gravity approach for defuzzification (5-7) [15]:

$$x_{CA} = \frac{\sum_{i=1}^n \mu_i(CA) x_i}{\sum_{i=1}^n \mu_i(CA)} \quad (5)$$

$$y_{p_c} = \frac{\sum_{i=1}^n \mu_i(p_c) y_i}{\sum_{i=1}^n \mu_i(p_c)} \quad (6)$$

$$z_{p_m} = \frac{\sum_{i=1}^n \mu_i(p_m) z_i}{\sum_{i=1}^n \mu_i(p_m)} \quad (7)$$

2.7. Elitism replacement with filtration

After the fuzzy crossover and mutation operators are applied, elitism replacement technique is used as the replacement strategy. The offspring have to compete with their parents in order to allow transition into the new population. In other words, fitter chromosomes will survive for the next generation and they are never lost unless better solutions are found. In the elitism replacement technique, both parent and offspring populations are considered together as a single population. Then this population is sorted in a non-increasing order of their associated fitness value and the first half of the chromosomes from this combined population are selected as the chromosomes for the new population of the next generation.

Table 3. Fuzzy rule base for CA and p_c and p_m

Rule	$T_{1,t}$	$T_{2,t}$	$T_{3,t}$	CA	p_c	p_m
1	Low	Low	Low	High	High	High
2	Low	Low	Medium	High	High	High
3	Low	Low	High	Medium	Medium	High
4	Low	High	Low	High	Medium	High
5	Low	High	Medium	Medium	Medium	Medium
6	Low	High	High	Medium	Low	Low
7	Medium	Low	Low	High	High	High
8	Medium	Low	Medium	Medium	Medium	Medium
9	Medium	Low	High	Medium	Medium	Low
10	Medium	High	Low	Medium	Medium	Medium
11	Medium	High	Medium	Medium	Medium	Low
12	Medium	High	High	Medium	Low	Low
13	High	Low	Low	High	High	High
14	High	Low	Medium	Medium	Medium	Medium
15	High	Low	High	Low	Medium	Low
16	High	High	Low	Low	Medium	Medium
17	High	High	Medium	Low	Low	Low
18	High	High	High	Low	Low	Low

In order to overcome of identical chromosomes, the filtration technique is used to add diversity to the new population. In this technique, one of the identical chromosomes is kept while the others are removed and replaced by new feasible chromosomes that are generated randomly. As the filtration procedure involves the process of “identify”, “re-generate” and “re-evaluate” of the new chromosomes, which requires a certain amount of computation time, it is sensible to just invoke the procedure every R generation (where R is a parameter, e.g. 100) or when there is at least 10 percent of the population that are identical [13].

3. P-MEDIAN PROBLEM

Facility location problems appeared in numerous applications such as communications, industrial transportation and distribution networks. One of the well-known facility location problems is the p -median problem which locates p facilities among n demand points and allocates the demand points to the facilities. The objective is to minimize the total demand-weighted distance between the demand points and the facilities. The following formulation of the p -median problem is introduced by Revelle and Swain [27], where (6-10):

$$\min \sum_{i=1}^n \sum_{j=1}^n w_i d_{ij} x_{ij} \quad (6)$$

$$s.t \sum_{j=1}^n x_{ij} = 1, \quad \forall ij \quad (7)$$

$$x_{ij} \leq y_{ij}, \quad \forall ij \quad (8)$$

$$\sum_{j=1}^n y_j = p \quad (9)$$

$$x_{ij} = 0 \text{ or } 1, \forall ij \text{ and } y_{ij} = 0 \text{ or } 1, \forall ij \quad (10)$$

where n =total number of demand point

$$x_{ij} = \begin{cases} 1 & \text{if point } i \text{ is assigned to facility location at point } j \\ 0 & \text{otherwise} \end{cases} \quad y_{ij} = \begin{cases} 1 & \text{if a facility is location at point } j \\ 0 & \text{otherwise} \end{cases}$$

d_{ij} = travel distance between point i, j
 P =number of facilities to be located

In a p -median problem, let the number of facilities is N , then these facilities are indexed to 1, 2, ..., N . We use a simple encoding where the genes correspond to the indexes of the selected facilities. For example, for capacitated 6-median facility location problem, (2, 5, 6, 23, 19, 7) is a chromosome where demand points 2, 5, 6, 23, 19 and 7 are selected as facility locations.

A population of chromosome solutions is generated to form an initial population. For our proposed algorithm the population is generated randomly. The population will be evaluated by fitness function. The fitness evaluation must be able to reflect the objective and direct the search towards optimal solution. The fitness of a chromosome is the same as the objective function value of the solution it corresponds to, and it can be calculated using the problem data.

4. COMPUTATIONAL EXPERIMENTS

4.1. Experimental design

A benchmark data set that is tested in this research for all algorithms inclusive of 40 PMFLPs proposed by Beasley [10] and was extensively utilized in the literature for the testing of PMFLPs algorithms. Each of the 40 instances in the class is a graph with a corresponding value for p . Every node is a customer and a potential facility, and the cost of assigning a customer to a facility is the length of the shortest path between the corresponding nodes. The number of nodes in this class varies from 100 to 900, and the value of p from 5 to 200.

The computational experiment runs reported here were performed by using a GA with initial population that is generated randomly. Regarding number of facilities and number of medians, the initial population are created by considering index for facilities. For instance, in the p -median problem where there are 50 facilities and $p=10$, the length of chromosomes is considered to be 10. These 10 numbers are selected from index 1 to 50 and the layout of them is not important.

The performance of these approaches is measured by the percentage deviation (D) between the benchmark problems taken from the ORLibrary (Beasley, [10] and the heuristic results from the proposed algorithms.

$$PD = \frac{|B_i - R_i|}{B_i} \times 100\% \quad (11)$$

where B_i is the benchmark result and R_i is the computational result obtained by the proposed algorithm.

The algorithms were coded in C++ and run on a Pentium IV with 2.0 GHz CPU and 2.0 GB of RAM and each problem instance was tested 30 times with a maximum 20 CPU seconds per run.

4.2. Computational results of comparison selection mechanisms

In this subsection, we present the computational results of the proposed sexual selection based on integer encoding that was mentioned literature. In order to scrutinise the performance and the ability of the proposed sexual selection, some commonly used selection mechanisms are considered. Common abbreviations of these methods are presented in Table 4.

Table 4. Common abbreviation of selection mechanism

Selection Mechanism	Abbreviation
Sexual Selection	SX
Roulette Wheel	RW
Tournament Selection	TS
Linear Ranking	LR
Stochastic Universal Sampling	SUS
Truncation Selection	TR

The computational experiments reported here were performed by using a standard GA with initial population that is created randomly. In this algorithm the population size of 100, Partially Mapped Crossover (PMX) as the default crossover operator with probability p_c , and pairwise mutation with probability p_m , are considered. After crossover and mutation, all offspring will be replaced their parent (en-bloc replacement). This replacement strategy is used because we want to assess the performance of the selection mechanisms. The stopping criterion for the proposed algorithm is the 20 CPU seconds.

To see whether the sexual selection provides a good coverage and to avoid the premature convergence, comparison results for some selection mechanisms are reported in the Figure 4 when the algorithms terminate at 20 CPU seconds per run. In order to further assess the results, these results are compared in two groups. Firstly, all algorithms are compared based on problem instances with a fixed median point i.e. short chromosome length. As can be seen in Figure 6, average percentage deviation for sexual selection is less than 1.5% whereas for other selection mechanisms average percentage deviations are at least 1.5%.

In the other comparison, we considered different demand points with maximum median points from Table 5. In this case, the lengths of the chromosomes are larger than other situation for each demand point. As can be seen in Figure 7, for all problems, results of the average percentage deviation for sexual selection are better than other selection mechanisms. The average percentage deviation for sexual selection is between 0.8% and 1.8% whereas for other selection mechanisms are between 1.2% and 3.5%.

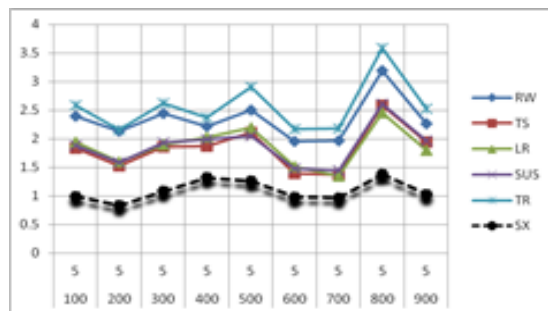


Figure 6. Comparison percentage deviations for selection mechanisms based on problem instances with fixed median point

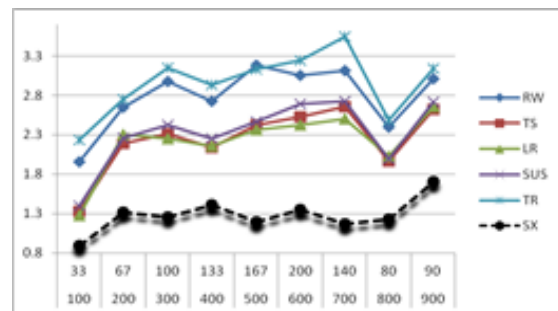


Figure 7. Comparison percentage deviations for selection mechanisms with maximum median points

4.3. Computational results of FGA based on crossover operator and probability selection technique

In this subsection, we present the computational results of the proposed integer encoding FGA based on crossover operator and probability selection technique named FGAC, for solving PMFLPs. In order to assess the performance and the ability and comparison of the proposed FGAC, some commonly used crossover operators in binary encoding are considered. Common abbreviations of these methods are presented in Table 5.

Table 5. Common abbreviations of crossover based on integer encoding

Crossover methods	Abbreviation
Partially Mapped Crossover	PMX
Order Crossover	OX
Cycle Crossover	CX
Position Based Crossover	PBX
Alternating Position Crossover	APX

The computational experiment reported here are performed by using a standard GAs (SGAs) with initial population that is created randomly. In this algorithm the population size of 100, tournament selection of size two and pairwise mutation with probability $p_m = 0.02$ are considered. The crossover operators listed in Table 4 are applied with a fixed crossover probability, $p_c = 0.70$ for SGAs. After the crossover and mutation, all offspring will be replaced their parent (en-bloc replacement). The stopping criterion for the proposed algorithms is the 20 CPU seconds.

In the other comparison results, we considered different demand points with maximum median points. In this case, the lengths of the chromosomes are larger than other situation for each demand point. As can be seen in Figure 8 the average percentage deviation for all group problems, results of the average percentage deviation for the proposed FGAC are better than the other crossover operators.

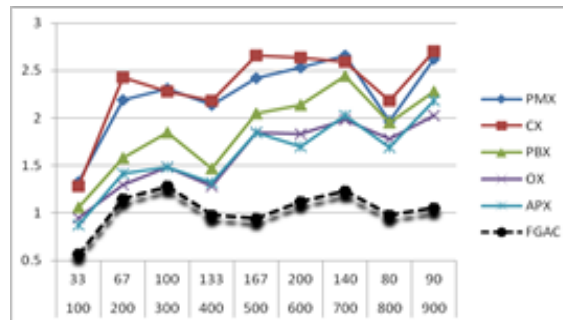


Figure 8. Comparison percentage deviations for crossover operators with maximum median points

4.4. Comparison of FGA with competitors

The same benchmark problem instances are used for the computational experiments. The numerical results are computed after making 30 independent runs for statistical significance. Each run is terminated if one of the following stopping conditions is met:

- Fitness value is not improved after 100 generations;
- CPU time is more than 200 CPU seconds.

The results acquired by the FGA point out that the proposed complete FGA is effective for PMFLPs, judging by the small percentage deviations shown. This can be partially verified by comparing the average percentage deviation columns given in Table 6, under Appendix. Finally, the ability of the FGA to generate the optimal solutions is demonstrated in the last column, in which the FGA is able to find optimal values for 37 out of 40 problem instances tested.

5. CONCLUSION

In this study, we proposed a fuzzy genetic algorithm using sexual selection, crossover and mutation operators and probabilities selection technique for solving the p -median facility location problems. Various techniques have been introduced into the proposed algorithm to further enhance the solutions quality. The proposed algorithm has been tested against the best known solutions reported in the literature, using problem instances from the benchmarks in [6]. The computational results and small percentage deviations for each problem showed that the proposed algorithm is competitive in terms of the quality of the solutions found.

REFERENCE

- [1] K. Wang, "A New Fuzzy Genetic Algorithm Based on Population Diversity," In Proceeding of the 2001 international Symposium on Computational Intelligence in Robotics and Automation, New Jersey, 108-112, 2001.
- [2] S. M. Odeh, A. M. Mora, M. N. Moreno, and J. J. Merelo, "A Hybrid Fuzzy Genetic Algorithm for an Adaptive Traffic Signal System," Hindawi Publishing Corporation, *Advances in Fuzzy Systems*, Article ID 378156, 11 pages, Volume 2015.
- [3] K. A. De Jong. Analysis of the Behavior of a Class of Genetic Adaptive systems. PhD Thesis, Department of Computer and Communication Sciences, University of Michigan (1975).
- [4] F. Herrera and M. Lozano, "Fuzzy genetic algorithms: issues and models," Technical Report DECSAI-98116, Department of Computer Science and A. I., University of Granada, (1998).
- [5] F. Herrera and M. Lozano, "Fuzzy adaptive genetic algorithm: design, taxonomy, and future directions. *Soft Computing*," 545-562, 7 (2005).
- [6] M Jalali Varnamkhasti, "A hybrid of adaptive neuro-fuzzy inference system and genetic algorithm," *Journal of Intelligent and Fuzzy Systems*, vol. 25, pp. 793-796, 2013.
- [7] M. Jalali Varnamkhasti and L. S. Lee, "A fuzzy genetic algorithm based on binary encoding for solving multidimensional knapsack problems," *Journal of Applied Mathematics*, vol. 2012, Article ID 703601, 23 pages, 2012.
- [8] Y. Song, G. Wang, P. Wang and A. Johns, "Environmental/Economic Dispatch Using Fuzzy Logic Controlled Genetic Algorithm," In: IEEE Proceeding on Generation, Transmission and Distribution. The Institution of Engineering and Technology, London, 377-382, 144 (1997).
- [9] Q. Li, D. Zheng, Y. Tang and Z. Chen, "A New Kind of Fuzzy Genetic Algorithm," *Journal of University of Science and Technology Beijing*, 85-89, 1 (2001).
- [10] C. Re Velle and R. Swain, "Central Facilities Location," *Geographical Analysis*, 30-42, 2 (1970).
- [11] C. H. Wang and T. P. Hong, "Integrating membership functions and fuzzy rule sets from multiple knowledge sources," *Fuzzy Sets System*, 141-154, 112 (2000).
- [12] H. Liu, Z. Xu and A. Abraham, "Hybrid Fuzzy-Genetic Algorithm Approach for Crew Grouping," In Nedjah N. Mourelle L. M., Vellasco M. M. B. R, Abraham A., Koppen M. (eds.), *Proceedings of the 2005 5th International Conference on Intelligence Systems Design and Applications*. IEEE Computer Society, Washington, DC, 332-337, (2005).
- [13] F. Herrera and M. Lozano, "Adaptive Control of the Mutation Probability by Fuzzy Logic Controllers," *Parallel problem Solving from Nature VI*, Berlin: Springer. 335-344, (1996).
- [14] F. Herrera, E. Herrera-Viedma, M. Lozano and J. L. Verdegay, "Fuzzy Tools to Improve Genetic Algorithms," In *Proceeding of the European Congress on Intelligent Techniques and soft Computing*, 1532-1539, (1994).
- [15] M. Jalali Varnamkhasti and L. S. Lee, "Fuzzy Genetic Algorithm with Sexual Selection," *Second International Conference and Workshops on Basic and Applied Sciences and Regional Annual Fundamental Science Seminar UTM, Malaysia*, 38-43, (2009).
- [16] S. A. Jafari, S. Mashohor and M. Jalali Varnamkhasti, "Committee neural networks with fuzzy genetic algorithm," *Journal of Petroleum Science and Engineering*, 217-223, (4) 76 (2011).
- [17] M. Jalali Varnamkhasti and L. S. Lee, "A Genetic Algorithm with Fuzzy Crossover Operator and Probability," *Advance Operation Research*, (2011).
- [18] A. X. Yand Y. X. Jin, "A Fuzzy C-Means Clustering Algorithm Based on Improved Quantum Genetic Algorithm," *Internationa Journal of Database Theory and Application*, Vol. 9, pp. 227-236, (2016).
- [19] E. Cam, G. Gorel, and H. Mamur, "Use of the Genetic Algorithm-Based Fuzzy Logic Controller for Load-Frequency Control in a Two Area Interconnected Power System," *Appl. Sci.*, 7(3), 308, 2017.
- [20] J. D. Schaffer, "A study of control parameters affecting online performance of genetic algorithms for function optimization," In *Proceeding Third International Conference Genetic Algorithm*, 51-60, (1989).
- [21] J. E. Beasley, "A note on solving large p-median problems," *European Journal of Operational Research*, 270-273, 21 (1985).
- [22] J. J. Grefenstette, "Optimization of control parameters for genetic algorithms," *IEEE Transactions System, Man, and Cybernetics*, 122-128, (1) SMC-16 (1986).
- [23] K. A. De Jong, "Adaptive system design: a genetic approach," *IEEE Transactions System, Man, and Cybernetics*, 566-574, (9) 10 (1980).
- [24] M. Mitchell, "An Introduction to Genetic Algorithms," (USA: MIT Press, 1996).
- [25] L. Davis, "Adapting operator probabilities in genetic algorithms," In *Proceedings of the Third International Conference on Genetic Algorithms and Their Applications*, Morgan Kaufmann, pp. 61-69 (1989).
- [26] C. Fernandes, R. Tavares and A. Rosa, "niGAVaPS – Outbreeding in Genetic Algorithms," presented at Evonet Summer School 99, Antwerp (Belgium), (1999).
- [27] E. S. Correa, M. T. A. Steiner, A. A. Freitas and C. Carnieri, "A genetic algorithm for solving a capacitated p-median problem," *Numerical Algorithm*, 373-388, 35 (2004).
- [28] F. Chiyoshi and R. Galv ao, "A Statistical Analysis of Simulated Annealing Applied to the P-Median Problem," *Annals of Operations Research*, 61-74, 96 (2000).
- [29] P. Hansen, N. Mladenovic and D. Perez-Brito, "Variable neighborhood decomposition search," *Journal of Heuristics*, 335-350, (3) 7 (2001).

APPENDIX

The values for comparison between FGA and local search algorithms as given Table 6.

Table 6. Comparison between FGA and local search algorithms

Test	n	p	Opt.	SA	VNDS	GA	FGA
1	100	5	5819	0.00	0.00	0.00	0.00
2	100	10	4093	0.00	0.29	0.00	0.00
3	100	10	4250	0.00	0.47	0.00	0.00
4	100	20	3034	0.00	0.00	0.00	0.00
5	100	33	1355	0.00	0.22	0.00	0.00
6	200	5	7824	0.00	0.00	0.00	0.00
7	200	10	5631	0.00	0.14	0.00	0.00
8	200	20	4445	0.00	0.20	0.08	0.00
9	200	40	2734	0.00	0.69	0.00	0.00
10	200	67	1255	0.00	0.32	0.00	0.00
11	300	5	7696	0.00	0.00	0.00	0.00
12	300	10	6634	0.00	0.00	0.00	0.00
13	300	30	4374	0.47	0.00	0.00	0.00
14	300	60	2968	0.35	0.03	0.00	0.00
15	300	100	1729	0.00	0.12	0.23	0.04
16	400	5	8162	0.00	0.00	0.00	0.00
17	400	10	6999	0.00	0.14	0.00	0.00
18	400	40	4809	0.27	0.04	0.00	0.00
19	400	80	2845	0.77	0.14	0.00	0.00
20	400	133	1789	0.11	0.00	0.17	0.00
21	500	5	9138	0.00	0.00	0.00	0.00
22	500	10	8579	0.00	0.00	0.00	0.00
23	500	50	4619	0.00	0.09	0.00	0.00
24	500	100	2961	0.14	0.11	0.03	0.00
25	500	167	1828	0.16	0.07	0.22	0.01
26	600	5	9917	0.00	0.04	0.00	0.00
27	600	10	8307	0.00	0.16	0.00	0.00
28	600	60	4498	0.09	0.20	0.02	0.00
29	600	120	3033	0.20	0.05	0.07	0.00
30	600	200	1989	0.15	0.00	0.40	0.02
31	700	5	10086	0.00	0.00	0.00	0.00
32	700	10	9297	0.00	0.06	0.00	0.00
33	700	70	4700	0.04	0.10	0.00	0.00
34	700	140	3013	0.00	0.00	0.07	0.00
35	800	5	10400	0.00	0.00	0.00	0.00
36	800	10	9934	0.00	0.54	0.02	0.00
37	800	80	5057	0.40	0.18	0.00	0.00
38	900	5	11060	0.00	0.10	0.00	0.00
39	900	10	9423	0.00	0.00	0.00	0.00
40	900	90	5128	0.16	0.12	0.10	0.00

SA=Simulated Annealing [21]

VNDS=The Variable Neighbourhood Decomposition Search[28]

GA=Genetic Algorithm [29]